

CS 60
FOURTH QUIZ
July 23, 2009

WRITE ALL YOUR ANSWERS ON SPACE PROVIDED.
ANSWER ALL FOUR QUESTIONS. TOTAL POINTS ARE 33.
YOU MAY ASSUME THAT EACH SECTION OF CODE BELOW IS IN ITS
OWN FILE WHEN COMPILED

NAME: _____

1 { Circle for each part True or False depending whether or not the statement is true or false. Each question is worth 1.5 Point }

- { True or **False** } When we execute in the `main` function the statement (declaration) `List t = m;` the assignment member function for Class List is invoked.
- { **True** or False } When we execute in the `main` function the statement (declaration) `List t(m);` the copy constructor for Class List is invoked.
- { True or **False** } When we execute in the `main` function the statements `List t; t = m;` the assignment member function for the Class List is not invoked.
- { **True** or False } In C++ it is valid to have the functions `int sqrt(int x);` and `float sqrt(float x);` defined one after the other.
- { True or **False** } It is possible to pass parameters by reference in C++ and in C.
- { True or **False** } In C++ it is valid to have the functions `int rand();` and `float rand();` defined one after the other, provides that the `int` one is defined before the `float` one.
- { True or **False** } In C++ the function declaration `bool XXX(Rest r, int Time=1, int Num=2);` means that every time `Time` has the value of 1 when we start executing the function, `Num` will have the value of 2 (at that time too).
- { True or **False** } In C++ an `inline` function means that the function must be defined in one line.
- { True or **False** } In C++ we use `new` to create local variables.

2 {Class Player}

Consider the following Class defined in class.

```
class Player {
private:
    string name;
    int games;
    int points;
    int rebounds;
    int assists;
public:
    Player();
    ~Player();
    Player(string);
    Player(const Player &);
    void Reset();
    void Print();
    bool AddGame(int, int, int);
    void SetName(string);
    void SetGames(int);
    void SetPoints(int);
    void SetRebounds(int);
    void SetAssists(int);
    string Name();
    int Games();
    int Points();
    int Rebounds();
    int Assists();
};
```

(a) [1 Point] Is the above declaration of **Class Player** incorrect because there are functions that do not have a return-type?

NO

(b) [1 Point] Are there any constructors in the above declaration? If so which functions are the constructors.

Player() and Player(string)

(c) [1 Point] Are there any destructors in the above declaration? If so which functions are the destructors.

Player()

(d) [1 Point] Are there any copy constructors in the above declaration? If so which functions are the copy constructors.

```
Player(const Player &)
```

3 {More Classes [5 Points]}

Consider the following code that uses class B defined below. (a) What does the program print? (b) Is there any memory leak? If so, where? (c) Which memory location is involved in the memory leak? (d) How many times does the destructor gets invoked when the following code is executed? (e) List the points in the main program where the destructor gets invoked.

```
#include <iostream>
using namespace std;
class B
{private: int i;
         int *j;
public:
    B(int a, int b) { i=a; j = new int; *j = b;}
    ~B() {}
    void prints() { cout << i << " " << *j << endl;}
};
int main()
{
    {B p(5,9);
      B * q = new B(3,6);
      p.prints();
      p = *q;
      p.prints();
      delete q;
    }
}
```

```
5 9
3 6
```

Yes, memory leaks at `p=*q` and `delete q`.

The memory allocated dynmaically (stored in `*j`).

twice (one for `*q` and the other for `p`)

Delete `q` and end of interior block.

4 {Pointers}

(a) [3 points] What does the following code print?

```
int xxx = 12;
int yyy = 4;
```

```
int *pi = &yyy;
pi = &xxx;
printf("%d\n",*pi);
*pi = 21;
printf("%d %d\n",xxx,yyy);
```

```
12
21 4
end{verbatim}
\vspace*{0.75in}
```

```
\newpage
\noindent
(b) [3 points]
What does the following code print?
```

```
\begin{verbatim}
int xxx = 7;
int yyy = 25;
int *pi = &yyy;
*pi = xxx;
printf("%d\n",*pi);
*pi = 10;
printf("%d %d\n",xxx,yyy);
```

```
7
7 10
```

5 {Basic Code}

a.- [2 points] Clearly indicate the value printed by the following code.

```
#include <iostream>
using namespace std;

main()
{
    int x = 1;
    x += 100;
    if (x = 100) cout << "YES" << endl;
    else cout << "NO" << endl;
}
```

YES

b- [2 Points]

```
#include <iostream>
using namespace std;
```

```
int main(void)
{
    int i=1;
        { i = 3;
            i++;
        }
    if (i > 2) cout << i << endl;
    else cout << "0" << endl;
}
```

4